Discrete-Euclidean operations

Gaëlle Largeteau-Skapin, Eric Andres

Discrete-Euclidean Operations

Gaëlle Largeteau-Skapin and Eric Andres

Laboratoire SIC,
Université de Poitiers,
BP 30179 86962 Futuroscope Chasseneuil cédex, France
{*glargeteau,andres*}*@sic.univ-poitiers.fr,*

Abstract

In this paper we study the relationship between the Euclidean and the discrete space. We study discrete operations based on Euclidean functions: the discrete smooth scaling and the discrete-continuous rotation. Conversely, we study Euclidean operations based on discrete functions: the discrete based simplification, the Euclidean-discrete union and the Euclidean-discrete co-refinement. These operations operate partly in the discrete and partly in the continuous space. Especially for the discrete smooth scaling operation, we provide error bounds when different such operations are chained.

Key words: discrete geometry, operations, multi-representation modeler.

1 Introduction

In computer imagery, the discrete and the Euclidean spaces are considered apart. Both spaces have different properties which led to separate branches in computer imagery: computer modeling and image synthesis on one side and computer vision and image analysis on the other side. Operations are primarily conducted in the Euclidean or the discrete space. Operations might be trivial in one space and difficult to transpose in the other one. For instance, there isn't a satisfying discrete rotation that is at the same time one-to-one and commutative, which are two fundamental properties of the continuous rotation. Boolean operations (intersection, union, difference) that are trivial in the discrete space become tedious to perform in the continuous space because of numerical errors.

In this paper we explore new types of operations: *discrete-Euclidean operations and Euclidean-discrete operations*. The idea behind these operations is to move to the best adapted space for any given operation. For instance, the

discrete-Euclidean rotation is a discrete operation that moves to the Euclidean space, performs the rotation there and then moves back to the discrete space. The Euclidean-discrete co-refinement is a Euclidean operation that moves to the discrete space, performs the boolean operations and moves back to the Euclidean space. Discrete-Euclidean and Euclidean-discrete operations are performed partly in the discrete and partly in the Euclidean space with help of discretization and continuation transforms. A discretization allows us to move from the continuous space to the discrete space. A continuation transform allows us to move from the discrete space to the continuous space. We introduce the term *continuation* to avoid the ambiguity of the term *reconstruction* used in very different settings. Of course, this works if the discretization and the continuation transforms are associated. We explain some of the properties that have to be verified by both the discretization and the continuation.

Operations that operate partly in the discrete and in the continuous world exist already. For instance, in image analysis, discrete space is often simply embed in continuous space in order to use the tools of continuous geometry. In the same way, discrete grids are often used as space subdivisions in computer modelling. The novelty here, with our operations, is the explicit use of the discretization and continuation steps. This allows us to propose new operations that haven't, to the authors bets knowledge, not been proposed before such as, for instance, the discrete-continuous scaling operation. It allows us also to control the error bounds when doing reverse operations.

The paper explores discrete-Euclidean and Euclidean-discrete transforms. In [1] we have introduced an original discrete-Euclidean operation based on the scaling transform in the discrete space: the discrete smooth scaling. The discrete smooth scaling describes a discrete object in a smaller (finer) grid. We want to perform this operation without filtering or smoothing. We want therefore to avoid interpolation or fuzzy operations. The discrete smooth scaling operation possesses a remarkable property: the almost stability by inverse scale. If we make a discrete smooth scale of factor $\alpha \geq 1$ followed by a discrete smooth scale of factor $\frac{1}{\alpha}$ we obtain the original discrete object with an error bounded by a factor proportional to $\frac{1}{\alpha}$. In this paper, we propose new results on the error bounds when several discrete smooth scales are chained. We show that the order in which we chain the discrete smooth scaling transforms with various scales has an influence on the error bounds. As a new result we also show that these error bounds are optimal. We introduce and discuss the properties of a new discrete-Euclidean operation: the discrete-Euclidean rotation. In a second part of the paper, we look into Euclidean-Discrete operations. We go over the discrete based simplification introduced in [1]. The operation consists, this time, starting with a continuous object, to discretize with a given grid size and then to reconstruct it. When we reconstruct a discrete object, the "shape complexity" (resulting vertex and edge number) depends on the size of the object. The smaller the object, the less complex the reconstructed object.

Very similarly to what happens for the discrete smooth scaling, the Hausdorff distance between the original object and the simplified object is bounded by a factor proportional to the grid size. We introduce the Euclidean-discrete union and the Euclidean-discrete co-refinement as first attempts to explore Euclidean-discrete boolean operations.

The interest of these operations is that they each make use of the properties of the other space. The discrete operation uses the properties of the continuous space and the continuous operation those of the discrete space. These operations show how the duality between the discrete and the continuous space can be used at our advantage.
In section two, we introduce the basic notions used in this paper and discuss the properties discretizations and continuations have to verify. In the third section we look into discrete-Euclidean operations and the error bounds when chaining the operations and there reverse. In section four we discuss in the same way Euclidean-discrete operations. In section five we present the implementation and illustrations of these operations. We conclude and present perspectives in the last section of the paper.

2 Preliminaries

2.1 Basic notations in discrete geometry

The following notations correspond to those given by Cohen and Kaufman in [2] and those given by Andres in [3]. We provide only a short recall of these notions.

A **discrete** (resp. **Euclidean**) **point** is an element of $\mathbb{Z}^n$ (resp. $\mathbb{R}^n$). A **discrete** (resp. **Euclidean**) **object** is a set of discrete (resp. Euclidean) points. We denote p_i the ith coordinate of a point p of $\mathbb{Z}^n$. The **voxel** $\mathbb{V}(p) \subset \mathbb{R}^n$ of a discrete nD point p is defined by $\mathbb{V}(p) = [p_1 - \frac{1}{2}, p_1 + \frac{1}{2}] \times ... \times [p_n - \frac{1}{2}, p_n + \frac{1}{2}]$. For a discrete object D, $\mathbb{V}(D) = \bigcup_{p \in D} \mathbb{V}(p)$. The distances we use in this paper are all designed by $d_k(a, b) = \sqrt[n]{|a_x - b_x|^n + |a_y - b_y|^n}$.

In this paper, we use the Hausdorff distance defined by:

Definition 1 *Let h be the direct Hausdorff distance: $A \subset \mathbb{R}^n$, $B \subset \mathbb{R}^n$, $h(A, B) = max_{a \in A} (min_{b \in B} (d_2(a, b)))$, where d_2 is the Euclidean distance. The Hausdorff distance between A and B is $H(A, B) = max(h(A, B), h(B, A))$.*

One of the major Euclidean operation we are going to use throughout the

paper is the Euclidean scaling operation Sc_α by a scale α. We consider that the center of the scaling operation is the center of the space. We have therefore for x a Euclidean point, $Sc_\alpha(x) = \alpha.x$.

This paper is based on the relations between the continuous and the discrete spaces and the way operations can benefit from this duality. For this we need discretizations and continuations that operate together.

2.2 Digitization and Continuation

In this paper we are dealing with operations that operate partially in the discrete space $\mathbb{Z}^n$ and partially in the continuous space $\mathbb{R}^n$. For this we need to be able to travel between both spaces with appropriate transforms. A transformation from the Euclidean to the discrete space is called a *discretization*. This is sometimes also called digitization. We prefer however the term discretization because we don't see the transform from the continuous space to the discrete space as simply a sampling of the continuous space but rather as a mathematical transform between two spaces. The transformation from the discrete space to the Euclidean space is classically called a *reconstruction*. The term reconstruction has however different meanings in the computer vision and computer graphics literature and that is why the authors prefer to introduce the term *continuation*.
Simply defining a discretization (resp. continuation) as a transform that maps $\mathbb{R}^n$ to $\mathbb{Z}^n$ (resp. $\mathbb{Z}^n$ to $\mathbb{R}^n$) is in our case not enough. Discretization and continuation in the case of discrete-continuous and continuous-discrete operations have to be associated if we want *meaningful* operations. More precisely, a continuation $\mathcal{R}$ that is associated to a discretization $\mathcal{D}$ has to verify several properties. It is easy to see that we can define an equivalence relation $\approx$ between two Euclidean objects E and F by $E \approx F$ iff $\mathcal{D}(E) = \mathcal{D}(F)$. There is a one-to-one mapping between the discrete objects and the equivalence classes defined by $\approx$. The continuation has to stay in the equivalence class when we discretize and then continuate (perform a continuation): $\mathcal{R}(\mathcal{D}(E)) \approx E$. Of course, in general, $\mathcal{R}(\mathcal{D}(E)) \neq E$.

In the framework of this paper, we consider discretizations that verify the following property: $\mathcal{D}(\mathcal{R}(A)) = A$ for a discrete object A. This property is verified in many practical situations and by most discretization methods. This property is indeed verified if there isn't any *missing information* in A. The property isn't verified, for instance, when we reconstruct the border of a region after an edge detection on a noisy image. We speak of *partial continuation* in this case. Partial continuation doesn't prevent from doing discrete-continuous operations but in this case the discussion on the error bounds aren't meaningful. We will consider in this paper that we aren't dealing with partial con-

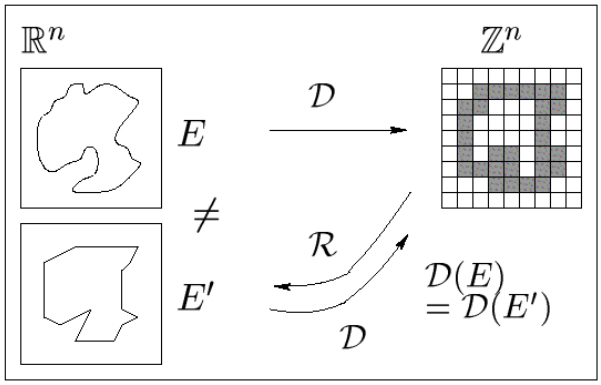

Fig. 1. Coherence between discrete and continuous spaces.

tinuations.

Let us recall [1] a class of discretization schemes that suit the purpose of this paper: the *narrow offset discretizations*. This corresponds to most known discretization transforms used in applications including Bresenham generation algorithms [4], the supercover model [2,5–7], the naïve discretization [8], the standard model [3], etc. In this paper we continue to consider mainly those kind of discretizations. It allows us to present classical discretization methods in a same framework. We needed also a geometric definition of a discretization scheme in order to perform our discussions on error bounds. The experiments presented in this paper have been conducted with the standard analytical model [3] (see also Fig. 2). A narrow offset discretization transform is based on a *narrow offset area*. A narrow offset area $\mathbb{O}$ is defined for classes of Euclidean objects. It simply has to verify two fundamental conditions: A narrow offset area $\mathbb{O}(E) \subset \mathbb{R}^n$ of a Euclidean object E must be *narrow* meaning that if $x \in \mathbb{O}(E) \cap \mathbb{Z}^n \Rightarrow \mathbb{V}(x) \cap E \neq \varnothing$. It requires the discretization of a Euclidean object E to be composed of voxels that are intersected by E. The second condition is a constructive condition. A narrow offset area must verify a stability property for the union: $\mathbb{O}(E \cup F) = \mathbb{O}(E) \cup \mathbb{O}(F)$.

Definition 2 *A narrow offset discretization based on a narrow offset area is defined by:*

$$\begin{aligned} \mathcal{D} : \mathbb{R}^n &\longrightarrow \mathbb{Z}^n \\ \mathcal{D}(E) &= \{p \in \mathbb{Z}^n \,|p \in \mathbb{O}(E)\} = \mathbb{O}(E) \cap \mathbb{Z}^n. \end{aligned}$$

A simple way to construct narrow offset discretizations is to define the offset area with a structuring element S centered at the origin. This defines the offset area by a simple Minkowski sum:

$$\mathbb{O}(E) = E \otimes S = \{e + s \,|e \in E, s \in S\}.$$

If we want the offset area to be narrow we have to verify $S \subseteq \left[-\frac{1}{2}, \frac{1}{2}\right] \times$

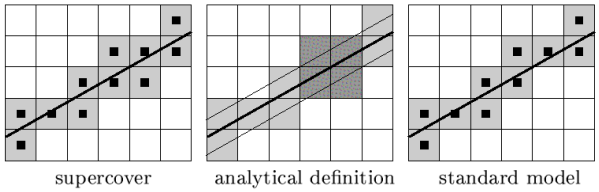

Fig. 2. Supercover and standard model examples.

$\cdots \times \left[-\frac{1}{2}, \frac{1}{2}\right]$. When $S = \left[-\frac{1}{2}, \frac{1}{2}\right] \times \cdots \times \left[-\frac{1}{2}, \frac{1}{2}\right]$ we have the supercover discretization [2,5–7].

An alternative way to construct narrow offset discretizations is to define the offset area with a distance d.

$$\mathbb{O}(E) = \left\{ x \in \mathbb{R}^n \,\middle|\, d(x, E) \leq \frac{1}{2} \right\}.$$

The construction with a distance is simply a particular case of the structuring element construction. If we take the unit ball $S = \left\{ x \in \mathbb{R}^n \,\middle|\, d(0, x) \leq \frac{1}{2} \right\}$ of the distance d as structuring element we obtain the same offset area.

The best known narrow offset discretization is called the supercover model [2,5–7] with an offset defined by the Manhattan distance d_∞. The distance d_1 defines the closed naïve model and the distance d_2 defines the closed Pythagorean model. All distances, of course, don't verify the narrowness property but many do. There exist also narrow offset areas that aren't defined with distances. This is the case for the Bresenham algorithms [4], the standard analytical model [3], the naïve discretization [8], etc.

Discretization based on narrow offset areas verify, by construction, properties such as $\mathcal{D}(E \cup F) = \mathcal{D}(E) \cup \mathcal{D}(F)$; $\mathcal{D}(E \cap F) \subset \mathcal{D}(E) \cap \mathcal{D}(F)$ and $E \subset F \implies \mathcal{D}(E) \subset \mathcal{D}(F)$. These properties ensure that we can build complex discrete objects out of a set of basic elements. We can, for instance, build all linear objects out of simplices.

The conditions we chose for our digitizations are a bit restrictive. The narrowness condition on the offset can under certain conditions be lifted if at least $E \subseteq \mathbb{O}(E)$. For instance, in the case of a discretization based on a distance, It is possible to consider $\mathbb{O}(E) = \left\{ x \in \mathbb{R}^n \,\middle|\, d(x, E) \leq \frac{\alpha}{2} \right\}$ where α is an arbitrary strictly positive value.

3 Discrete-Euclidean operations

In this section, we will study discrete operations from $\mathbb{Z}^n$ to $\mathbb{Z}^n$ that operate partially in the continuous space. In [1], we have presented the discrete smooth scaling, a discrete-Euclidean operation based on the scaling transform. In this paper, we provide new results on the error bound when several discrete smooth scalings and their reverse are chained. As a new result, we also show that the bound is optimal. These new results provide a new insight on how continuation and discretization error can be controlled. We introduce also a new discrete-continuous operation: the discrete-continuous rotation.

3.1 The discrete smooth scaling operation

The first discrete-Euclidean operation that we have proposed in [1] is called *discrete smooth scaling*. The idea behind this operation is to increase the size of a discrete object, by a factor $\alpha \in \mathbb{Z}^n$, while keeping a sharp border without smoothing, interpolating nor filtering (see Fig. 3). The scaling part in the transform is performed in the best adapted space: the Euclidean space.

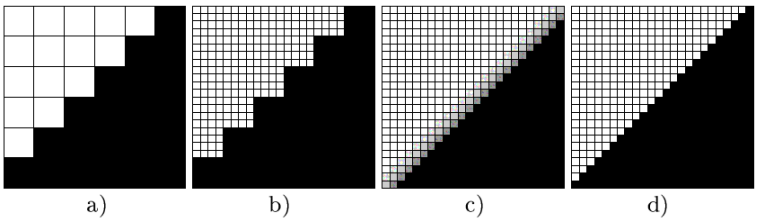

Fig. 3. a) original discrete object. b)reduced grid size. c) classical smoothing. d) discrete smooth scaling.

Definition 3 *[1] We call discrete smooth scaling of a discrete object A of $\mathbb{Z}^n$ by a scale α (denoted Sc_α), $\alpha \in \mathbb{R}^{+*}$, the following operation denoted $DSS_\alpha(A)$:*

$$DSS_\alpha(A) = \mathcal{D} \circ Sc_\alpha \circ \mathcal{R}(A).$$

One of the question that naturally comes to mind is the question of the reversibility of this operation. We showed in [1] that $DSS_{\frac{1}{\alpha}} \circ DSS_\alpha$ is not the identity however the error when performing a smooth scaling followed by its reverse smooth scaling is bounded and depends on both the dimension of the space and the scaling factor.

For a discrete object A, we note $A_{first} = \mathcal{R}(A)$ the continuation of the original discrete object and we note $A_{last} = Sc_{\frac{1}{\alpha}}(\mathcal{R}(DSS_\alpha(A)))$ the Euclidean object which discretization is $DSS_{\frac{1}{\alpha}}(DSS_\alpha(A))$. The error measure is a bound on the Hausdorff distance between both objects. It is actually not the difference

between the discrete object A and the discrete object $DSS_{\frac{1}{\alpha}}\left(DSS_{\alpha}\left(A\right)\right)$ that we characterize but the Hausdorff distance between the Euclidean objects A_{first}, just after the first continuation, and A_{last}, just before the last discretization. The difference between two discrete objects is indeed difficult to characterize in a meaningfull way. For instance, let us consider four Euclidean objects T, U, V and W. The fact that $H\left(T,U\right) < H\left(V,W\right)$ doesn't change the fact that we can have $\mathcal{D}(T) \cap \mathcal{D}(U) = \emptyset$ and $\mathcal{D}(V) = \mathcal{D}(W)$. If we consider that the grid is randomly positioned compared to the Euclidean object, for $\alpha < \beta$, there will be usually less difference between $DSS_{\frac{1}{\beta}}\left(DSS_{\beta}\left(A\right)\right)$ and A than between $DSS_{\frac{1}{\alpha}}\left(DSS_{\alpha}\left(A\right)\right)$ and A. That explains why we use this error measure.

Theorem 4 *[1] For a discrete object A, we note $A_{first} = \mathcal{R}(A)$ and A_{last} the Euclidean object which discretization is $DSS_{\frac{1}{\alpha}}\left(DSS_{\alpha}\left(A\right)\right)$.*
The Hausdorff distance between $A_{first} = \mathcal{R}(A)$ and A_{last} is bounded by:

$$H\left(A_{first}, A_{last}\right) = H\left(\mathcal{R}\left(A\right), Sc_{\frac{1}{\alpha}}\left(\mathcal{R}\left(\mathcal{D}\left(Sc_{\alpha}\left(\mathcal{R}(A)\right)\right)\right)\right)\right) \leq \frac{1}{\alpha}\sqrt{n}.$$

Corollary 5 *[1]* $\lim_{\alpha\to\infty} H\left(A_{first}, A_{last}\right) = 0$.

Theorem 4 shows that the error bound between the objects just after the discretization and just before the continuation are inversely proportional to the scaling factor. The greater the scale factor, the "better" the reversibility properties of the discrete smooth scaling transform become. This can be useful, as we will see for the Euclidean-discrete co-refinement (see section 4.3). The proof of theorem 4 and corollary 5 can be found in [1]. The error bounds presented in [1] and in this paper are based on the supercover discretization [2,5–7]. The standard discretization, in our implementation of the discrete-continuous and continuous-discrete operations, has the same error bound as the supercover because the narrow offset areas are almost identical. Indeed, the offset area of the supercover discretization is a closed hypercube of size 1 (the unit ball for the distance d_{∞}). The offset area of the standard discretization is a hypercube of size 1 open on some sides (to avoid bubbles in the discretizations. See [3] for more details). That explains why there is a factor $\sqrt{n}$ in all the bounds we propose in [1] and in this paper. The offset area of the supercover discretization is the *largest* possible offset area of all narrow offset discretizations. The bounds we propose are therefore the worst case that can be encountered within the conditions we set for our discretizations.

Now, let us study more precisely the discrete smooth scaling properties when it is combined with other DSS operations. We present a new result on the error measure with the succession of discrete smooth scalings and the related reverse smooth scalings: let $(\alpha_i)_{i\in[1..k]}$ $(\forall i \in [1,k],\ \alpha_i \in \mathbb{R})$ be a set of scaling coefficient.

Lemma 6 *For a discrete object A, we note $A_{first} = \mathcal{R}(A)$ and A_{last} the last Euclidean object before the end result, which discretization is*

$$\mathbb{O}(A_{last}) \cap \mathbb{Z}^n = DSS_{\alpha_k} \circ ... \circ DSS_{\alpha_1}(A).$$

For an Euclidean point x in A_{first}, the corresponding point in A_{last} is in the worst case the Euclidean point $\left(\Pi_{i=1}^{k}\alpha_i\right)x + \left(\sum_{i=2}^{k}\left(\Pi_{j=i}^{k}\alpha_j\right)+1\right)\sqrt{n}$. (1)

Proof:
Let x be the starting Euclidean point ($\in \mathcal{R}(A)$). First we perform a scale transform with coefficient α_1. We therefore obtain $\alpha_1 x$. Then we apply a discretization followed by a continuation. In the worst case, we get $y = \alpha_1 x + \sqrt{n} - \epsilon$, with ϵ arbitrarily small (for a supercover discretization we can have $\epsilon = 0$). Therefore the result of $\mathcal{R} \circ \mathcal{D} \circ Sc_{\alpha_1}$ is in the worst case the Euclidean point: $\alpha_1 x + \sqrt{n}$. Combining this result with $\mathcal{R} \circ \mathcal{D} \circ Sc_{\alpha_2}$ leads to: $\alpha_1\alpha_2 x + \alpha_2\sqrt{n} + \sqrt{n} = \alpha_1\alpha_2 x + (\alpha_2 + 1)\sqrt{n}$. After all the DSS, we obtain in the worst case the Euclidean point: $\left(\Pi_{i=1}^{k}\alpha_i\right)x + \left(\sum_{i=2}^{k}\left(\Pi_{j=i}^{k}\alpha_j\right)+1\right)\sqrt{n}$. □

Theorem 7 *For a discrete object A, we note $A_{first} = \mathcal{R}(A)$ and A_{last} the last Euclidean object before the end result, which discretization is*

$$\mathbb{O}(A_{last}) \cap \mathbb{Z}^n = \overbrace{DSS_{\frac{1}{\alpha_1}} \circ ... \circ DSS_{\frac{1}{\alpha_k}}}^{reverse\ discrete\ smooth\ scaling} \circ \overbrace{DSS_{\alpha_k} \circ ... \circ DSS_{\alpha_1}}^{discrete\ smooth\ scaling}(A).$$

The Hausdorff distance between A_{first} and A_{last} is bounded by:

$$H(A_{first}, A_{last}) \leq \left(\frac{1}{\Pi_{i=1}^{k}\alpha_i} + 2\sum_{i=1}^{k-1}\frac{1}{\Pi_{j=1}^{i}\alpha_j}\right)\sqrt{n}.$$

Moreover, this bound is optimal: it is the smallest upper bound.

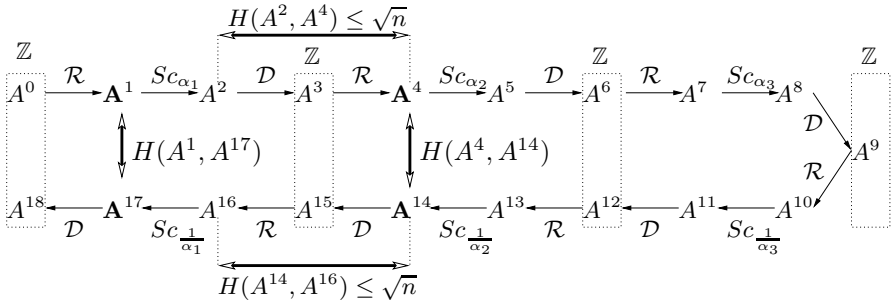

Fig. 4. Example of theorem 7 for $k = 3$.

Proof:
Because of the constructive property of the narrow offset areas, it is sufficient to prove the result for a point. Let x be the starting Euclidean point ($\in \mathcal{R}(A)$). After all the DSS, we get the Euclidean

point $\left(\Pi_{i=1}^{k}\alpha_i\right)x + \left(\sum_{i=2}^{k}\left(\Pi_{j=i}^{k}\alpha_j\right)+1\right)\sqrt{n}$ (see Lemma 6). The first reverse DSS leads to: $\left(\Pi_{i=1}^{k-1}\alpha_i\right)x + \left(\frac{\sum_{i=2}^{k}\left(\Pi_{j=i}^{k}\alpha_j\right)}{\alpha_k}+\frac{1}{\alpha_k}+1\right)\sqrt{n}$ $= \left(\Pi_{i=1}^{k-1}\alpha_i\right)x + \left(\sum_{i=2}^{k-1}\left(\Pi_{j=i}^{k-1}\alpha_j\right)+\frac{1}{\alpha_k}+2\right)\sqrt{n}$. The next reverse DSS (with α_{k-1}) result is: $\left(\Pi_{i=1}^{k-2}\alpha_i\right)x + \left(\sum_{i=2}^{k-2}\left(\Pi_{j=i}^{k-1}\alpha_j\right)+\frac{1}{\alpha_k\alpha_{k-1}}+\frac{2}{\alpha_{k-1}}+2\right)\sqrt{n}$. Once all the reverse DSS are applied, we obtain: $x + \left(\frac{1}{\Pi_{i=1}^{k}\alpha_i}+2\sum_{i=1}^{k-1}\frac{1}{\Pi_{j=1}^{i}\alpha_j}\right)\sqrt{n}$. The difference between the starting point x and its transform is therefore in the worst case: $\left(\frac{1}{\Pi_{i=1}^{k}\alpha_i}+2\sum_{i=1}^{k-1}\frac{1}{\Pi_{j=1}^{i}\alpha_j}\right)\sqrt{n}$. □

Remark: On figure 4 we can see that $H(A^1, A^{17}) = \frac{1}{\alpha_1}\left(2\sqrt{n} + H(A^4, A^{14})\right)$. We therefore have a recursive definition of the error bound.

If we compare the error bound provided by theorem 7 to the error that occurs by the direct DSS_α with $\alpha = \Pi_{i=1}^{k}\alpha_i$, we can see that the direct operation is much more efficient since it is bounded by $\frac{1}{\Pi_{i=l-1}^{k}\alpha_i}\sqrt{n}$ (see theorem 4).

A very important remark on this error measure is that it depends on the $(\alpha_i)_{i\in[1,k]}$ order: the error bound with $\alpha_1 = 2$ and $\alpha_2 = 5$ (≈ 1.55) is not the same as the error bound with $\alpha_1 = 5$ and $\alpha_2 = 2$ (≈ 0.70) (see illustrations section 5.2 fig. 12). The best order to perform this operation (i.e. the one that minimizes the bound measure) is $\forall i \in [1, k[, \alpha_i \geq \alpha_{i+1}$.

The error bound in the general case (the reverse DSS are not necessarily processed in the same order as the DSS) is given by the following theorem:

Theorem 8 *Let $(\alpha_i)_{i\in[1,k]}$ be the ordered list of DSS coefficients and let $(\alpha'_i)_{i\in[1,k]}$, $\forall i \in [1,k]\ \exists j \in [1,k]$ such that $\alpha'_i = \alpha_j$, be the ordered list of reverse DSS coefficients. For a discrete object A, we note $A_{first} = \mathcal{R}(A)$ and A_{last} the Euclidean object which discretization is*

$$\mathbb{O}\left(A_{last}\right) \cap \mathbb{Z}^n = \overbrace{DSS_{\frac{1}{\alpha'_1}} \circ ... \circ DSS_{\frac{1}{\alpha'_k}}}^{reverse\ discrete\ smooth\ scaling} \circ \overbrace{DSS_{\alpha_k} \circ ... \circ DSS_{\alpha_1}}^{discrete\ smooth\ scaling}(A)\,.$$

The Hausdorff distance between A_{first} and A_{last} is bounded by:

$$H\left(A_{first}, A_{last}\right) \leq \left(\sum_{i=1}^{k}\frac{1}{\Pi_{j=1}^{i}\alpha'_j} + \frac{\sum_{i=2}^{k}\Pi_{j=i}^{k}\alpha_j}{\Pi_{i=1}^{k}\alpha'_i}\right)\sqrt{n}.$$

Proof:
Let x be the starting Euclidean point ($\in \mathcal{R}(A)$). Again, because of the

constructive property of the narrow offset areas, it is sufficient to prove the result for a point. The result of all the DSS is given by Lemma 6: $\left(\Pi_{i=1}^{k}\alpha_i\right)x + \left(\sum_{i=2}^{k}\left(\Pi_{j=i}^{k}\alpha_j\right)+1\right)\sqrt{n}$. The first reverse DSS with coefficient $\frac{1}{\alpha'_k}$ leads to: $\frac{\left(\Pi_{i=1}^{k}\alpha_i\right)}{\alpha'_k}x + \left(\frac{\sum_{i=2}^{k}\left(\Pi_{j=i}^{k}\alpha_j\right)}{\alpha'_k} + \frac{1}{\alpha'_k} + 1\right)\sqrt{n}$. Once all the reverse DSS are applied, the result is: $\left(\frac{\Pi_{i=1}^{k}\alpha_i}{\Pi_{i=1}^{k}\alpha'_i}\right)x + \left(\frac{\sum_{i=2}^{k}\Pi_{j=i}^{k}\alpha_j}{\Pi_{i=1}^{k}\alpha'_i} + \sum_{i=1}^{k}\frac{1}{\Pi_{j=1}^{i}\alpha'_j}\right)\sqrt{n}$. Since $\forall i \in [1,k]\ \exists j \in [1,k]$ such that $\alpha'_i = \alpha_j$, $\frac{\Pi_{i=1}^{k}\alpha_i}{\Pi_{i=1}^{k}\alpha'_i} = 1$. We obtain the Euclidean point: $x + \left(\frac{\sum_{i=2}^{k}\Pi_{j=i}^{k}\alpha_j}{\Pi_{i=1}^{k}\alpha'_i} + \sum_{i=1}^{k}\frac{1}{\Pi_{j=1}^{i}\alpha'_j}\right)\sqrt{n}$. The difference between the starting point x and its transform is therefore: $\left(\frac{\sum_{i=2}^{k}\Pi_{j=i}^{k}\alpha_j}{\Pi_{i=1}^{k}\alpha'_i} + \sum_{i=1}^{k}\frac{1}{\Pi_{j=1}^{i}\alpha'_j}\right)\sqrt{n}$. □

Let us complete this study about the error bounds by determining the error bound if we combine DSS operations and then use the direct reverse $DSS_{\frac{1}{\alpha}}$ operation with $\alpha = \Pi_{i=1}^{k}\alpha_i$ (see figure 5).

Theorem 9 *For a discrete object A, we note $A_{first} = \mathcal{R}(A)$ and A_{last} the Euclidean object which discretization is*

$$\mathbb{O}\left(A_{last}\right) \cap \mathbb{Z}^n = \overbrace{DSS_{\frac{1}{\Pi_{i=1}^{k}\alpha_i}}}^{reverse\ discrete\ smooth\ scaling} \circ \overbrace{DSS_{\alpha_k} \circ ... \circ DSS_{\alpha_2} \circ DSS_{\alpha_1}}^{discrete\ smooth\ scaling}(A).$$

The Hausdorff distance between A_{first} and A_{last} is bounded by:

$$H\left(A_{first}, A_{last}\right) \leq \left(\sum_{i=1}^{k}\frac{1}{\Pi_{j=1}^{i}\alpha_j}\right)\sqrt{n}.$$

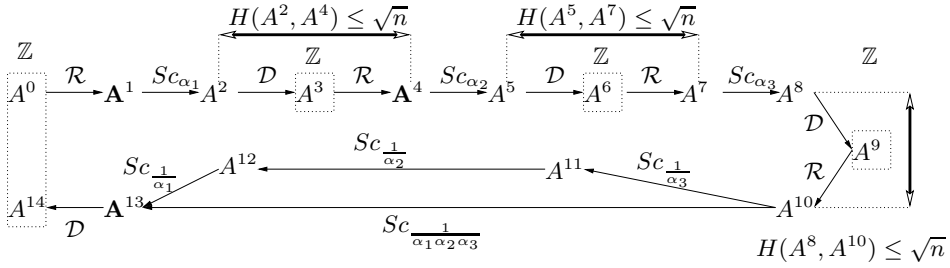

Fig. 5. Example of direct inversion of DSS combination with $k = 3$.

Proof:
Figure 5 shows the intuitive proof of the theorem with $k = 3$. Let x be the starting Euclidean point ($\in \mathcal{R}(A)$). The result of all the DSS is given by Lemma 6: $\left(\Pi_{i=1}^{k}\alpha_i\right)x + \left(\sum_{i=2}^{k}\left(\Pi_{j=i}^{k}\alpha_j\right)+1\right)\sqrt{n}$. The reverse DSS with coefficient $\frac{1}{\Pi_{i=1}^{k}\alpha_i}$ leads to: $x + \left(\frac{\sum_{i=2}^{k}\left(\Pi_{j=i}^{k}\alpha_j\right)+1}{\Pi_{i=1}^{k}\alpha_i}\right)\sqrt{n} = x + \left(\sum_{i=1}^{k}\frac{1}{\Pi_{j=1}^{i}\alpha_j}\right)\sqrt{n}$. The difference between the starting point x and its transform is therefore:

$\left(\sum_{i=1}^{k} \frac{1}{\Pi_{j=1}^{i}\alpha_j}\right)\sqrt{n}$. □

We can define a similar result for the operation consisting of a direct discrete smooth scaling (with $\alpha = \Pi_{i=1}^{k}\alpha_i$) and the combination of the reverse discrete smooth scaling $\left(DSS_{\frac{1}{\alpha_i}}\right)_{i\in[1,k]}$.

Theorem 10 *For a discrete object A, we note $A_{first} = \mathcal{R}(A)$ and A_{last} the Euclidean object which discretization is*

$$\mathbb{O}(A_{last}) \cap \mathbb{Z}^n = \overbrace{DSS_{\frac{1}{\alpha_1}} \circ ... \circ DSS_{\frac{1}{\alpha_k}}}^{reverse\ discrete\ smooth\ scaling} \circ \overbrace{DSS_{\Pi_{i=1}^{k}\alpha_i}}^{discrete\ smooth\ scaling} (A).$$

The Hausdorff distance between A_{first} and A_{last} is bounded by:

$$H(A_{first}, A_{last}) \leq \left(\sum_{i=1}^{k} \frac{1}{\Pi_{j=1}^{i}\alpha_j}\right)\sqrt{n}.$$

Proof:
Let x be the starting Euclidean point ($\in \mathcal{R}(A)$). The result of the DSS with coefficient $\Pi_{i=1}^{k}\alpha_i$ is: $\left(\Pi_{i=1}^{k}\alpha_i\right)x + \sqrt{n}$. The first reverse DSS leads to: $\left(\Pi_{i=1}^{k-1}\alpha_i\right)x + \left(\frac{1}{\alpha_k}+1\right)\sqrt{n}$. After all the reverse DSS we obtain: $x + \left(\sum_{i=1}^{k} \frac{1}{\Pi_{j=1}^{i}\alpha_j}\right)\sqrt{n}$. The difference between the starting point x and its transform is therefore: $\left(\sum_{i=1}^{k} \frac{1}{\Pi_{j=1}^{i}\alpha_j}\right)\sqrt{n}$. □

3.2 Discrete-Euclidean Rotation

After the study of the scale operation, we have decided to study another classical Euclidean operation: the rotation. We want to define a discrete rotation that takes advantage of the properties of the Euclidean rotation. This discrete rotation consists in a continuation to get an Euclidean object, than the rotation itself in the Euclidean space (the best one to perform a rotation) and finally, a discretization to obtain a discrete result. More formally:

Definition 11 *We call discrete-Euclidean rotation of a discrete object A of $\mathbb{Z}^n$ with an angle α and a center $\mathcal{C}$, $\alpha \in \mathbb{R}^{+*}$, $\mathcal{C} \in \mathbb{R}^n$ the following operation denoted $dR_{\alpha,\mathcal{C}}(A)$:*

$$dR_{\alpha,\mathcal{C}}(A) = \mathcal{D} \circ rot_{\alpha,\mathcal{C}} \circ \mathcal{R}(A).$$

Where $rot_{\alpha,\mathcal{C}}$ is the classical Euclidean rotation operation of angle α and center $\mathcal{C}$.

The intuitive $dR_{-\alpha,\mathcal{C}}$ is actually not an exact inverse operation but we can estimate the error that occurs by combining $dR_{-\alpha,\mathcal{C}}$ with $dR_{\alpha,\mathcal{C}}$. This error directly comes from the continuation part of the operation (see fig.7 and proof of theorem 4 in [1]).

Theorem 12 *For a discrete object A, we note $A_{first} = \mathcal{R}(A)$ and A_{last} the Euclidean object which discretization is $\mathbb{O}(A_{last}) \cap \mathbb{Z}^n = dR_{-\alpha,\mathcal{C}} \circ dR_{\alpha,\mathcal{C}}(A)$. The Hausdorff distance between A_{first} and A_{last} is bounded by:*

$$H(A_{first}, A_{last}) \leq \sqrt{n}.$$

Proof:(see figure 6)
Let x be the starting Euclidean point ($\in \mathcal{R}(A)$). Applying $\mathcal{R} \circ \mathcal{D} \circ rot_{\alpha,\mathcal{C}}$ leads in the worst case to: $rot_{\alpha,\mathcal{C}}(x) + \sqrt{n}$. Then the result of the reverse rotation is $rot_{-\alpha,\mathcal{C}}(rot_{\alpha,\mathcal{C}}(x) + \sqrt{n})$. Since rotation doesn't affect distances we obtain: $rot_{-\alpha,\mathcal{C}}(rot_{\alpha,\mathcal{C}}(x)) + \sqrt{n} = x + \sqrt{n}$. The difference between the starting point x and its transform is therefore $\sqrt{n}$. □

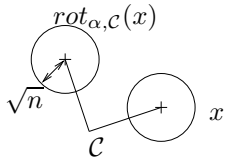

Fig. 6. Rotation reversibility.

The error measure with a succession of discrete-continuous rotations and their related reverse rotations is bounded by:

Theorem 13 *For a discrete object A, we note $A_{first} = \mathcal{R}(A)$ and A_{last} the Euclidean object which discretization is*

$$\mathbb{O}(A_{last}) \cap \mathbb{Z}^n = dR_{-\alpha_1,\mathcal{C}} \circ ... \circ dR_{-\alpha_k,\mathcal{C}} \circ dR_{\alpha_k,\mathcal{C}} \circ ... \circ dR_{\alpha_1,\mathcal{C}}(A).$$

The Hausdorff distance between A_{first} and A_{last} is bounded by:

$$H(A_{first}, A_{last}) \leq (2k-1)\sqrt{n}.$$

Proof:
Let x be the starting Euclidean point ($\in \mathcal{R}(A)$). The first rotation leads, in the worst case, to $rot_{\alpha_1,\mathcal{C}}(x) + \sqrt{n}$. Next rotation leads to $rot_{\alpha_1+\alpha_2,\mathcal{C}}(x) + 2\sqrt{n}$. Whatever the order we use to perform the rotations, we obtain $rot_{\sum_{i=1}^{k} \alpha_i,\mathcal{C}}(x) + k\sqrt{n}$ (see A^7 on figure 7). The first reverse rotation leads to $rot_{\sum_{i=1}^{k-1} \alpha_i,\mathcal{C}}(x) + (k+1)\sqrt{n}$. Finally, when all the reverse rotations

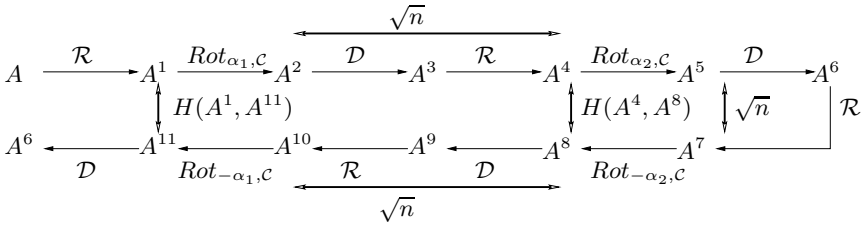

Fig. 7. Example of rotation reversibility with $k = 2$.

are performed, we have: $x + (k + (k-1))\sqrt{n}$. The difference between the starting point x and its transform is therefore $(2k-1)\sqrt{n}$. □

Remark: we also have a recursive definition of $H(A_{first}, A_{last})$, on figure 7 we can see that $H(A_{first}, A_{last}) = H(A^1, A^{11}) = H(A^4, A^8) + 2\sqrt{n}$.

4 Euclidean-discrete operations

In this section, we study several operations linking the Euclidean and the discrete spaces. These are operations from $\mathbb{R}^n$ to $\mathbb{R}^n$ that use the discrete operations properties. In [1], we presented a Euclidean-discrete operation based on the scaling transform (the discrete based simplification). Here, we present some additional results and we study two other operations : union and corefinement.

4.1 Discrete based simplification

This operation acts on a Euclidean object that is first discretized on a given grid size and then continuated. According to the grid size, details are gathered in the same voxel and thus do not appear in the continuated object [1]. The bigger the voxel, the lesser details from the Euclidean object will remain after the continuation. The object is simplified and can be represented at different levels of details (see Fig. 8). In fact, it is not the voxel size that changes but the object size. The object is scaled with the Euclidean scaling function to fit the grid size. For a scaling factor α the voxel size is $\frac{1}{\alpha}$.

Definition 14 *[1] We call discrete based geometrical simplification of a Euclidean object E of $\mathbb{R}^n$ by a factor α, $\alpha \in \mathbb{R}^{+*}$, the following operation denoted $Sp_\alpha(E)$:*

$$Sp_\alpha(E) = Sc_{\frac{1}{\alpha}} \circ \mathcal{R} \circ \mathcal{D} \circ Sc_\alpha(E).$$

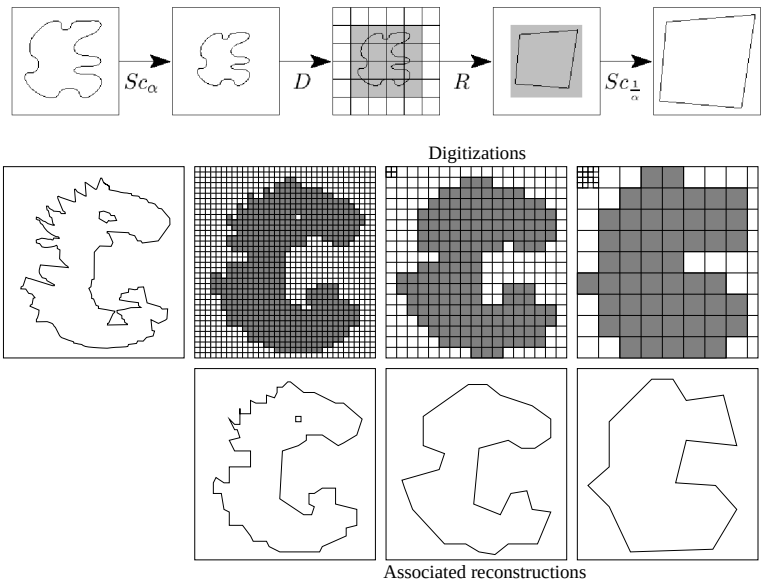

Fig. 8. Discrete based geometrical simplification principle : $\alpha = 1, \frac{1}{2}$ and $\frac{1}{4}$ corresponding to voxel sizes 1, 2 and 4.

For the discrete smooth scaling we gave a bound on the error that occurs when doing a smooth scaling and then its reverse. The discrete based simplification is actually very similar to the discrete smooth scaling and its reverse. We perform indeed a scaling of factor α, a discretization, continuation and a scaling of factor $\frac{1}{\alpha}$. It is therefore not very surprising that, as we proved in [1], the Hausdorff distance between an object E and its discrete based simplification Sp_α is bounded by a similar value than the one we obtained for the discrete smooth scaling.

Theorem 15 *[1] The Hausdorff distance between a Euclidean object E and its discrete based simplification $Sp_\alpha(E)$ is bounded by:*

$$\forall E \subset \mathbb{R}^n,\ H(E, Sp_\alpha(E)) \leq \frac{1}{\alpha}\sqrt{n}.$$

The reader can find the proof of this theorem in [1]. An interesting property of the discrete based simplification is that a combination of two simplifications of same coefficients is stable.

property: $\forall \alpha \in \mathbb{R},\ Sp_\alpha \circ Sp_\alpha = Sp_\alpha$.
Proof:
$Sp_\alpha \circ Sp_\alpha = \left[Sc_{\frac{1}{\alpha}} \circ \mathcal{R} \circ \mathcal{D} \circ Sc_\alpha\right] \circ \left[Sc_{\frac{1}{\alpha}} \circ \mathcal{R} \circ \mathcal{D} \circ Sc_\alpha\right]$
$= Sc_{\frac{1}{\alpha}} \circ \mathcal{R} \circ \mathcal{D} \circ \left[Sc_\alpha \circ Sc_{\frac{1}{\alpha}}\right] \circ \mathcal{R} \circ \mathcal{D} \circ Sc_\alpha$
$= Sc_{\frac{1}{\alpha}} \circ \mathcal{R} \circ \mathcal{D} \circ \mathbf{Id} \circ \mathcal{R} \circ \mathcal{D} \circ Sc_\alpha$
$= Sc_{\frac{1}{\alpha}} \circ \mathcal{R} \circ [\mathcal{D} \circ \mathcal{R}] \circ \mathcal{D} \circ Sc_\alpha$
$= Sc_{\frac{1}{\alpha}} \circ \mathcal{R} \circ \mathbf{Id} \circ \mathcal{D} \circ Sc_\alpha$

$= Sc_{\frac{1}{\alpha}} \circ \mathcal{R} \circ \mathcal{D} \circ Sc_\alpha \qquad (\mathcal{D} \circ \mathcal{R} = \mathbf{Id}$ but $\mathcal{R} \circ \mathcal{D} \neq \mathbf{Id})$
$= Sp_\alpha$. □

In applications, object simplification is rather seen as a progressive operation that starts out with a *complex* Euclidean object and then simplifies it more and more. There are two ways of doing this. The first method consists in applying decreasing scaling coefficients $\alpha_1 > \alpha_2 > \ldots > \alpha_k$ to the same starting object E. We obtain a sequence of simplified objects $Sp_{\alpha_1}(E)$, $Sp_{\alpha_2}(E)$, $\cdots$, $Sp_{\alpha_k}(E)$. The difficulty, with this approach, is that it is difficult to compare the topology of $Sp_{\alpha_i}(E)$ and $Sp_{\alpha_j}(E)$ for any i, j. The topology isn't simplified progressively as we would expect it. It can actually be rather erratic. Controlling the topology of the simplified objects to obtain a progressively simplified topology that goes with a progressive simplified geometry is difficult to achieve with this method. A second approach consists in simplifying the object by chaining discrete based geometrical simplifications. This way we are sure that a topological feature that disappears at one level of the simplification won't reappear later on. The topological simplification is more progressive this way. However the following result shows there are also some problems with this approach.

Theorem 16 *The Hausdorff distance between a Euclidean object E and its multiple discrete based simplification is bounded by:*

$$\forall(\alpha_i)_{i\in[1,k]},\ H(E, Sp_{\alpha_k} \circ ... \circ Sp_{\alpha_2} \circ Sp_{\alpha_1}(E)) \leq \left(\sum_{i=1}^{k} \frac{1}{\alpha_i}\right)\sqrt{n}.$$

Proof:
We know from theorem 15 that $H(E, Sp_{\alpha_1}(E)) \leq \frac{1}{\alpha_1}\sqrt{n}$ and that $H(Sp_{\alpha_{k-1}} \circ ... \circ Sp_{\alpha_1}(E), Sp_{\alpha_k} \circ ... \circ Sp_{\alpha_1}(E)) \leq \frac{1}{\alpha_k}\sqrt{n}$. By induction we obtain immediately the expected result. □

In the worst case, the error bound can diverge when we chain discrete simplifications. This reflects the fact that we can have geometrical deformations. Globally we can control the size of the simplified object by, for instance, bounding it by the bounding box size of the original object. This avoids global divergences but not local deformations that aren't progressive and that don't correspond to what you would expect from a progressive geometrical simplification. As we see, we have topological problems with the first approach and geometrical problems with the second approach. How we could define a discrete based geometrical simplification method that controls the topology and the geometry during the simplification remains unclear. There is an extensive literature related to mesh simplification that use discrete grids as space subdivision [9]. We need now to look into these methods in order to relate them

to our proposed Euclidean-discrete simplification operation.

4.2 Euclidean-discrete Union

Boolean operations are tedious to perform in the Euclidean space because of numerical problems while they are natural in the discrete one. With the following example of Euclidean-discrete operation, we explore a first boolean operation with our approach : the Euclidean-discrete union (see figure 9).

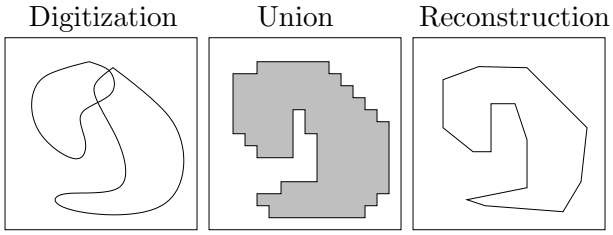

Fig. 9. Discrete based Euclidean union.

Definition 17 *We call Euclidean-discrete union of two Euclidean objects E and E' of $\mathbb{R}^n$ the following operation denoted $EDu\,(E, E')$:*

$$EDu\,(E, E') = \mathcal{R}\,(\mathcal{D}(E) \cup \mathcal{D}(E'))\,.$$

Where $\cup$ is the classical discrete union operation.

This operation has a very basic definition as we can see. It is not very useful in applications as such. It is interesting in a more global setting when performing Euclidean boolean operations that are sensitive to numerical problems. Discrete space allows a local control of the error (errors remain localized to a grid cell) and avoids these errors to propagate through a mesh as it can happen in several methods of co-refinement for instance [10–13].

4.3 Euclidean-discrete co-refinement

A very important operation in computer modelling in Euclidean space is the co-refinement operation. Co-refinement algorithms permit to build the space subdivision corresponding to the composition of several subdivisions (see figure 10). Several co-refinement algorithms used in modern Computer-Aided Design are presented in [10–13]. Most of these algorithms are based on basic boolean operations. For the same reason as for the Euclidean-discrete union, we perform the co-refinement in the best adapted space: the discrete space.

Definition 18 *We call Euclidean-discrete co-refinement of two Euclidean objects E and E' of $\mathbb{R}^n$ the following operation denoted $EDco\,(E, E')$ (see figure*

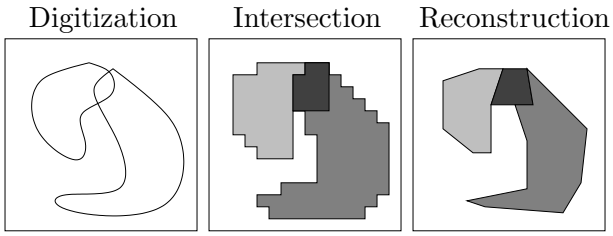

Fig. 10. Discrete based Euclidean co-refinement.

10): we note $\overline{\mathcal{D}(E)}$ *the complementary of* $\mathcal{D}(E)$.

$$EDco\,(E,E') = \left\{ \mathcal{R}\left(\mathcal{D}(E) \cap \overline{\mathcal{D}(E')}\right), \mathcal{R}\left(\mathcal{D}(E) \cap \mathcal{D}(E')\right), \mathcal{R}\left(\mathcal{D}(E') \cap \overline{\mathcal{D}(E)}\right) \right\}.$$

Where $\cap$ *is the classical discrete intersection operation.*

On figure 10 we can see the subdivision resulting of the co-refinement operation on two object E and E'. The result is in three part: $\mathcal{R}\left(\mathcal{D}(E) \cap \overline{\mathcal{D}(E')}\right)$ (in light grey), $\mathcal{R}\left(\mathcal{D}(E) \cap \mathcal{D}(E')\right)$ in dark grey and $\mathcal{R}\left(\mathcal{D}(E') \cap \overline{\mathcal{D}(E)}\right)$ in regular grey.

The precision of this operation depends (like all the operations we have already presented) on the grid size and is bounded by $\sqrt{n}$. One way of increasing the precision of the operation is to use the Euclidean scale operation. Using a scale operation is like decreasing the voxel size (see section 3.1. discrete smooth scaling) and thus improving the precision of the discrete operation. We define the precise co-refinement operation in the following way :

Definition 19 *We call Precise Euclidean-discrete co-refinement of parameter* α *of two Euclidean objects* E *and* E' *of* $\mathbb{R}^n$ *the following operation denoted* $EDco_\alpha\,(E,E')$*:* $EDco\,(E,E') = \Big\{ Sc_{\frac{1}{\alpha}}\Big(\mathcal{R}\left(\mathcal{D}(Sc_\alpha(E)) \cap \overline{\mathcal{D}(Sc_\alpha(E'))}\right)\Big),$
$Sc_{\frac{1}{\alpha}}\Big(\mathcal{R}\left(\mathcal{D}(Sc_\alpha(E)) \cap \mathcal{D}(Sc_\alpha(E'))\right)\Big), Sc_{\frac{1}{\alpha}}\Big(\mathcal{R}\left(\mathcal{D}(Sc_\alpha(E')) \cap \overline{\mathcal{D}(Sc_\alpha(E))}\right)\Big) \Big\}.$

This operation is an adaptation of the discrete smooth scaling operation (combined with its reverse) on a set of discrete objects. The error bound of this operation is therefore $\frac{1}{\alpha}\sqrt{n}$ (see theorem 4, page 8). The greater α, the better the precision of the operation. Of course, a bigger α value means also bigger Euclidean $Sc_\alpha(E)$ to discretize and bigger discrete objects $\mathcal{D}(Sc_\alpha(E'))$ to continuate. The increase in precision comes at cost in computation time.

5 Results: Implementation and Illustrations

Let us comment our implementation choices and present some images to illustrate the operations. The theoretical results we presented in this paper are valid in dimension n for a large class of discretizations and related continuation transforms. We implemented the operations in 2D.

5.1 Implementation

For several years our discrete geometry team develops a multi-representation modelling software intending to represent objects under four different embeddings (see Fig. 11): a discrete 2D pixel or 3D voxel representation, the region representation, its analytical equivalent and finally a a Euclidean representation [14]. This allows us to choose the best adapted representation form depending on the kind of operations we want to achieve.

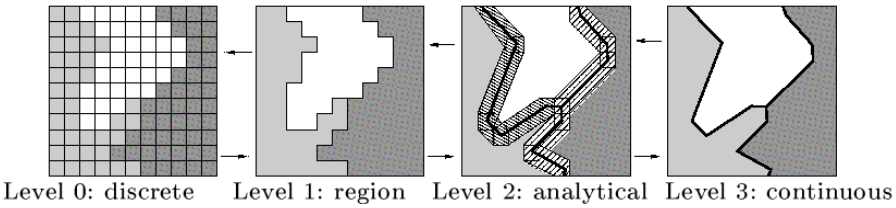

Fig. 11. Multi-representation modeler.

In this modeler discrete object are defined using the standard analytical model [3]. The continuation implemented in the modeler was defined in [15,16] and is based on the *preimage* notion [17]. This algorithm computes the set of Euclidean hyperplane segments which discretization contains the original discrete object: $\mathcal{R}(A) \subset \mathbb{V}(A)$ (the standard model is a cover). This approach is based on discrete analytical geometry and is composed of two steps: the recognition of discrete analytical hyperplane segments (see [18] for an overview on recognition algorithms) and the analytical polygonalization of the curve [19,16].

5.2 Illustrations

Here we present illustrations of the discrete smooth scaling transform combination. The first images concern the result of the DSS transform on the discretization of an octahedra with scaling factors $\alpha_1 = 1.25$ and $\alpha_2 = 5$. We remark that there is no error generation during the $DSS_{\frac{1}{5}} \circ DSS_5$ operation. This is of course particular to this object and these scales. The error bound

in this case is less than or equal to $\frac{1}{5}\sqrt{2} \approx 0.28$ (theorem 4). Combining this operation with another DSS transform leads to some errors this time. The error bound for the global transform $DSS_{0.8} \circ DSS_{0.2} \circ DSS_5 \circ DSS_{1.25}$ is: $\left(\frac{1}{1.25*5} + 2\frac{1}{1.25}\right)\sqrt{2} \approx 1.35$ (theorem 7). Errors are presented on figure 12 in light grey (not in the original object) and dark grey (in the original object but not in the result).

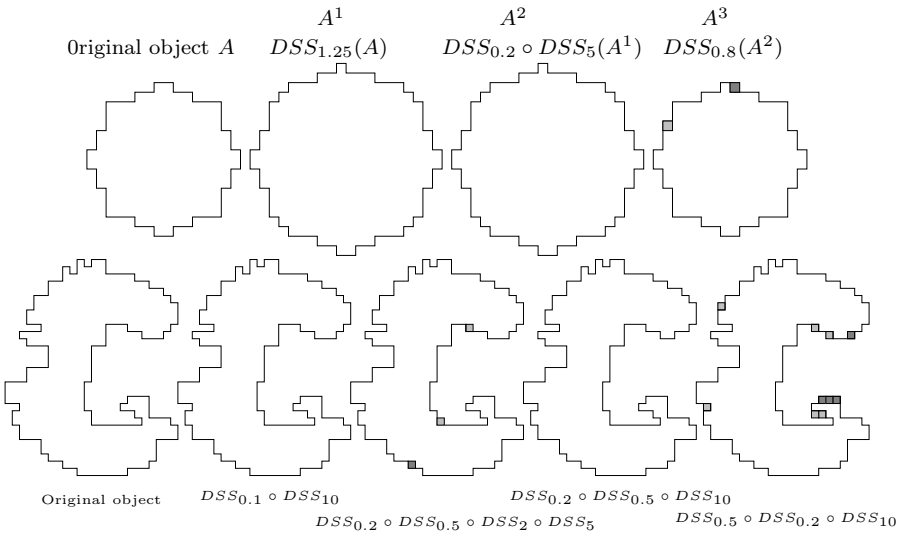

Fig. 12. Discrete smooth scaling combination examples.

The second example of Discrete Smooth Scaling operation result concerns a more detailled object. The figure presents the difference between the DSS_{10} and the combination of DSS_2 and DSS_5. The first result is exactly the same as the original object. The error bound is nonetheless not equal to zero since it is $\frac{\sqrt{2}}{10} \approx 0.14$ (theorem 4). The second result is obtained by the operation $DSS_{0.2} \circ DSS_{0.5} \circ DSS_2 \circ DSS_5$ ($\alpha_1 = 5$ and $\alpha_2 = 2$), we can see errors in light and dark grey. The error in this case is bounded by $\left(\frac{1}{2*5} + 2\frac{1}{5}\right)\sqrt{2} \approx 0.70$ (theorem 7). If we exchange the values and consider $\alpha_1 = 2$ and $\alpha_2 = 5$, we get an error close to 1.55. The next example is a combination of a direct DSS with $\alpha = 10$ and two reverse DSS ($\alpha_1 = \frac{1}{5}$ and $\alpha_2 = \frac{1}{2}$). No error appears on the discrete resulting object. The error in this case is bounded by: $\left(\frac{1}{2*5} + \frac{1}{5}\right)\sqrt{2} \approx 0.42$ (theorem 10). The last case is obtained by the inversion of the two reverse DSS and leads to an error of $\left(\frac{1}{2*5} + \frac{1}{2}\right)\sqrt{2} \approx 0.84$. As we can see, the number of wrong discrete points tends to increase when the error bound gets bigger which is to be expected.

The second discrete-Euclidean operation presented is the discrete-Euclidean rotation. The figure 13 presents some results on a simple square face and on a more detailed discrete object.

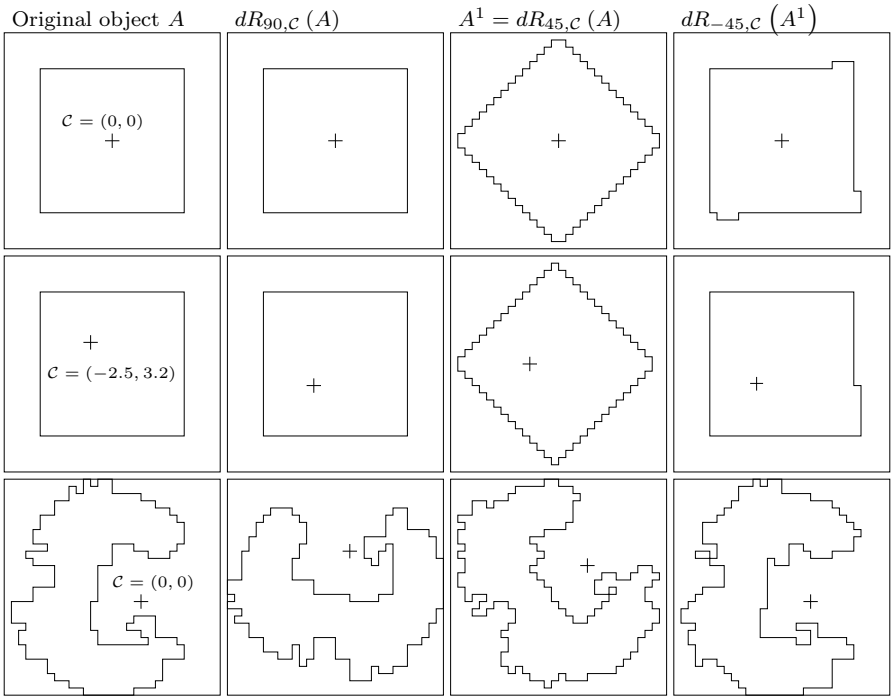

Fig. 13. Discrete-Euclidean rotation examples.

We can see that the rotation with $\alpha = 90$ is exact (that is what we expect) but this rotation is not reversible for all angle values. The rotation with $\alpha = -45$ is not the exact inverse of the rotation with angle $\alpha = 45$.

Illustrations for the discrete based simplification are available in [1].

6 Conclusion

In this paper, we have presented operations that use Euclidean and discrete space properties. They operate partly in the Euclidean and partly in the discrete space. In order to work, we need discretization and continuation transforms that are associated and that have properties that we tried to explore. This work is not complete and needs further investigations. We gave some classes of discretizations and continuations that allow the design of discrete-Euclidean and Euclidean-discrete operations. The criteria we gave cover the classical discretizations schemes such as Bresenham, Supercover, standard, naive models. What other classes of discretizations and continuations might also allow such operations needs to be looked at.

We did an extensive study on the error bounds when discrete smooth scalings and their reverse are combined. This is important because in all discrete-Euclidean and Euclidean-discrete operations, a scaling operation can be added so as to change the error bounds of the operation as we saw for the precise

Euclidean-discrete co-refinement.

We also studied the discrete-Euclidean rotation and proposed an optimal error bound for this operation when it is combined to its reverse and when it is chained to other discrete-Euclidean rotations. We think both operations (discrete smooth scaling and discrete-Euclidean rotation) could be very useful in pattern recognition: two images may represent the same object with different sizes and/or orientations. Using our operations, we can re-orient and re-scale images so that a comparison can be made easier.

The second part of our study is about Euclidean-discrete operations, mainly on boolean operations that are trivial in the discrete space and very useful in Euclidean CAD but difficult to perform because of numerous numerical errors.

The fundamental question is the question of the relations between the discrete and Euclidean space. In applications, we hope to apply this new insight in multi-level topological structure operations or on multi-scale described objects.

References

[1] G. Largeteau-skapin, E. Andres, Two discrete-euclidean operation based on the scaling transform, in: Discrete Geometry for Computer Imagery, LNCS, Szeged, Hungary, 2006, pp. 41–52.

[2] D. Cohen-Or, A. Kaufman, Fundamentals of surface voxelization, Graphical Models and Image Processing 57 (6) (1995) 453–461.

[3] E. Andres, R. Menon, R. Acharya, C. Sibata, Discrete linear objects in dimension n : The standard model, Graphical Models 65 (2003) 92–111.

[4] J. Bresenham, Algorithm for computer control of a digital plotter, IBM Systems Journal 4 (1) (1965) 25–30.

[5] E. Andres, R. Acharya, C. Sibata, The supercover 3d polygon, in: Discrete Geometry for Computer Imagery, LNCS, Lyon, France, 1996, pp. 237–242.

[6] E. Andres, P. Nehlig, J. Françon, Supercover of straight lines, planes and triangles, in: Discrete Geometry for Computer Imagery, Vol. 1347 of LNCS, Springer Verlag, 1997, pp. 243–257.

[7] C. Lincke, W. Wuthrich, Towards a unified approach between digitization of linear objects and discrete analytical objects, in: WSCG Int. Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, V. Skala (Ed.), 2000, pp. 124–131.

[8] J.-P. Reveillès, Géométrie discrète, calcul en nombres entiers et algorithmique, Ph.D. thesis, Université Louis Pasteur, Strasbourg, France (1991).

[9] P. Heckbert, M. Garland, Survey of polygonal surface simplification algorithms, in: Siggraph'97, Course Notes, no. 25, ACM Press, 1997.

[10] D. Cazier, J.-F. Dufourd, Rewriting-based derivation of eficient algorithms to build planar subdivisions, in: 12th Spring Conference on Computer Graphics, 1996, pp. 45–54.

[11] M. de Berg, O. Devillers, K. Dobrindt, O. Schwarzkopf, Computing a single cell in the overlay of two simple polygons, Information Processing Letters 63 (4) (1997) 215–219.

[12] M. Rivero, F. R. Feito, Boolean operations on general planar polygons, Computer & Graphics 24 (6) (2000) 881–896.

[13] S. Brandel, S. Schneider, M. Perrin, N. Guiard, J.-F. Rainaud, P. Lienhardt, Y. Bertrand, Automatic building of structured geological models, Journal of Computing & Information Science & Engineering 5 (2) (2005) 138–148.

[14] E. Andres, R. Breton, P. Lienhardt, Spamod: design of a spatial modeler, in: Digital and Image Geometry, advanced lectures, Vol. 2243 of LNCS, Springer Verlag, 2001, pp. 90–107.

[15] R. Breton, I. Sivignon, F. Dupont, E. Andres, Towards an invertible euclidean reconstruction of a discrete object, in: Discrete Geometry for Computer Imagery, Vol. 2886 of LNCS, Springer Verlag, Napoli, Italy, 2003, pp. 246–256.

[16] M. Dexet, E. Andres, Linear discrete line recognition and reconstruction based on a generalized preimage, in: Int. Workshop on Combinatorial Image Analysis, Vol. 4040 of LNCS, Springer Verlag, Berlin, Germany, 2006, pp. 174–188.

[17] L. Dorst, A. Smeulders, Discrete representation of straight lines, IEEE Transactions on Pattern Analysis and Machine Intelligence 6 (4) (1984) 450–463.

[18] A. Rosenfeld, R. Klette, Digital straightness, in: Int. Workshop on Combinatorial Image Analysis, Electronic Notes in Theoretical Computer Science, 2001, pp. 1–32.

[19] J. Françon, J.-M. Schramm, M. Tajine, Recognizing arithmetic straight lines and planes, in: Discrete Geometry for Computer Imagery, LNCS, Springer-Verlag, 1996, pp. 141–150.